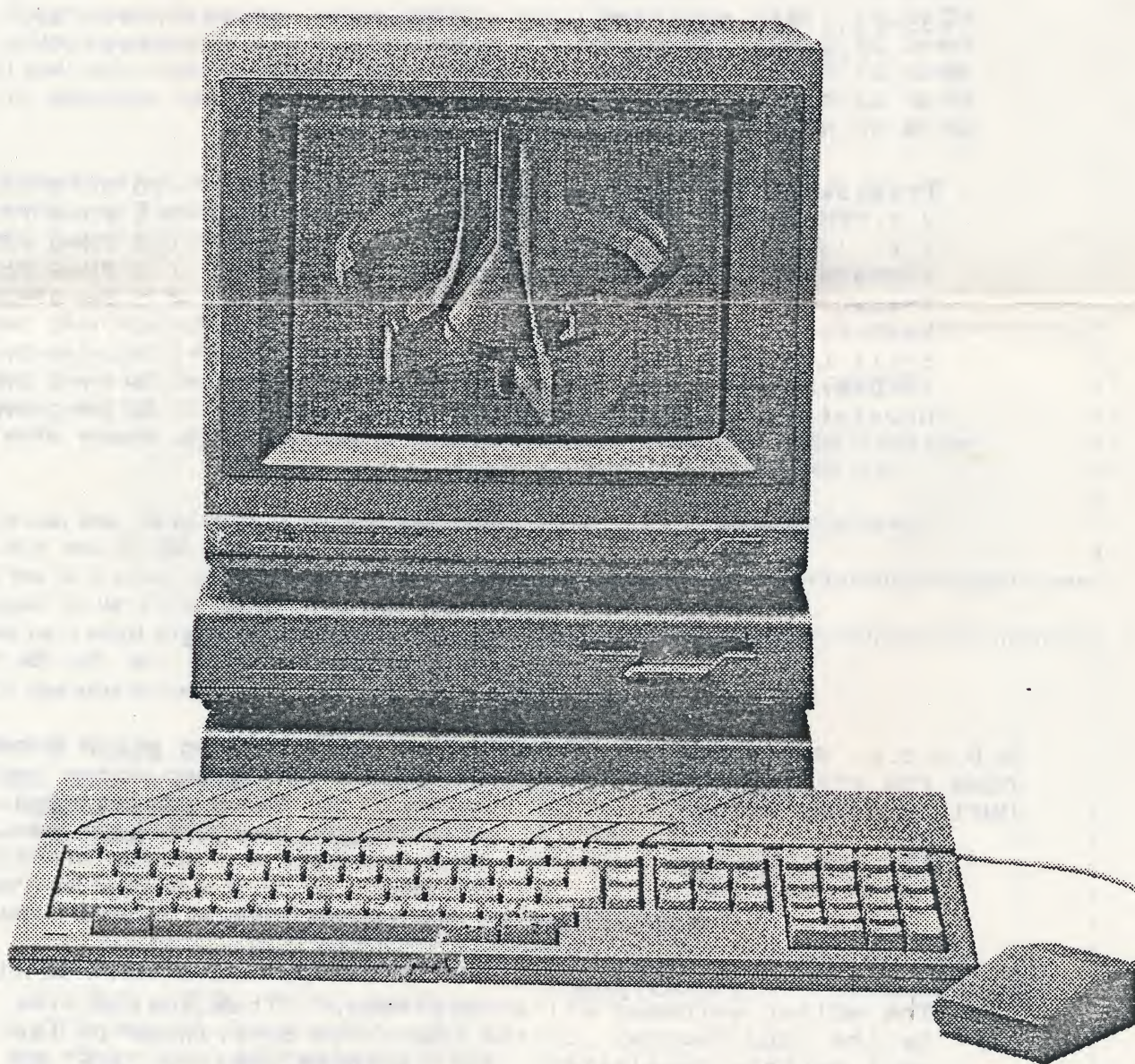


The

# I/O Connector

*The newsletter for the San Diego Atari Computer Enthusiasts*

February 88





## THE SAN DIEGO ATARI COMPUTER ENTHUSIASTS

(S.D.A.C.E.) is an independent, non-profit organization and user group with no connection to Atari Corporation. Membership includes access to the the program library, subscription to the I/O Connector, and access to other club activities. Permission to reprint articles from this newsletter in non-commercial form is permitted without specific authorization as long as original credit is granted.

### COMMERCIAL ADVERTISING RATES

\$35-Full \$30-Back \$20-Half \$10-Quarter \$5-Business Card  
(Prepaid Ads-pay for 5 months, get the 6th free!)

### S.D.A.C.E. Officers

President	Mark Lawless	691-7844	(10 AM-5 PM)
V.P. (8-bit)	Ron Miller	748-7195	( 5 PM-9 PM)
V.P. (ST)	Roger Fairchild	565-1078	( 5 PM-9 PM)
Secretary	David Delgadillo	475-6790	( 5 PM-9 PM)
Treasurer	Tom Andert	287-4198	( 5 PM-9 PM)
Membership	Dick Hiatt	541-7034	( 5 PM-9 PM)
8-bit Library	David Becker	280-1330	( 5 PM-9 PM)
ST Library	Mike Odegard	287-9282	( 5 PM-9 PM)
Newsletter	Tom Briant	224-8975	( 5 PM-9 PM)

### CORRESPONDENCE ADDRESS:

SAN DIEGO ATARI COMPUTER ENTHUSIASTS  
P.O. BOX 203076  
SAN DIEGO, CA 92120

S.D.A.C.E. 8-BIT BBS  
DOWN FOR REPAIRS  
UNTIL FURTHER NOTICE

S.D.A.C.E ST BBS  
SYSOP: RICK DEHAVEN  
(619) 284-3821  
300/1200/2400 BAUD 24 HOURS

### SUBMISSIONS TO THE NEWSLETTER

The editor welcomes all submissions. They must arrive by the 2nd Tuesday of the month to be printed in the next month's newsletter. Mail printed copy or 3.5" ST format disks with return postage to the club P.O. Box, or upload to one of the S.D.A.C.E. bulletin boards.



# THE PURSUIT OF HAPPINESS

by Tom Briant

The I/O Connector has 2 original articles! Please see Dave Delgadillo's article on PC-Ditto, where he explains how to use this package, and Paul Gardner's views on shareware.

Multiple font word processors should finally start appearing. MS-Write is at dealers now, Neutron Inc. will bring out WordUp, and a German word processor, Signum, should arrive real soon now.

If Spectrum's 512 colors don't satiate your chromatic desires, GFA Artist has 1024 colors. It requires a megabyte, but dealers have it NOW. Eidersoft, though, will bring out Quantum Paintbox with a full 4096 colors. I'm very curious about this one.

The ST also has better CAD packages now available. DRAFIX 1, a top IBM package, has now come out for the ST. I hope someone out there will submit design sketches for JACKSTAR 1, Atari's orbiting factory for manufacturing cheap, plentiful 1 billion byte memory chips.

On the home-brewed front, Mark Lawless will bring in his home-built hard disk to the ST meeting on February 15th. I hope someone will bring in their own 5.25" disk drive, too.

I kept my word and put the ST trackball conversion article in this issue. D.E. Venzelburger of S.P.A.C.E. wrote the original article. I found the joystick extension cords at Computer Outlet's Mission Gorge store. Radio Shack says they discontinued this item. If you find one there, GRAB IT!!! Otherwise, you will have to buy an IBM monitor extension cable at Computerland for substantial bucks.

On the other hand, you may prefer to keep your hands on the keyboard while using 1ST WORD. I advise you then to download 1ST MACRO.ARC from the ST bulletin board. Then purchase the Winter 1987 START magazine for the STARTkey Accessory on the disk. These two will enable you to control 1ST WORD from your keyboard. For those with 1ST WORD PLUS, well, the world now needs a macro file to control all those extra features.

On the 8-bit side, has anyone tried Virtuoso, which merges graphics, text, music, and animation? Please see the brief review from the Portland Atari Club newsletter; but I'd prefer some hands-on reviews from S.D.A.C.E. members. Also, WordPerfect says that if 10,000 8-bit users write in for WordPerfect on the 8-bit Atari, they'll port WordPerfect to the 8-bit Atari. Probably look a lot like Apple II WordPerfect, the only 8-bit version available.

Finally, yes, it's true. Vierd, but true. Larry Flynt Publishing will publish ANALOG and ST-LOG. They won't change editorial content, nor will the notorious Mr. Flynt include ads for HUSTLER in the back pages. Think of this as a vote of confidence in the Atari computers. Mr. Flynt only cares if ANALOG and ST-LOG have profit potential, so he must believe Atari has a future.

## LISTING OF SAN DIEGO AREA ATARI BBSs

Area code 619, 300/1200 BAUD, 24 hours unless otherwise noted

Code	Name	Computer	Baud	Number
4 !	ST-SDACE	ST	3/12	284-3821
1	Sherwood Forest	8-bit/ST	3/12	276-5603
4	ST-MIDI Connection	ST	2400	452-7535
5 !	Computer Blvd.	ST	2400	589-0565
4 *	Computer Outlet	ST	2400	282-6815
6	The Lord's House	8-bit	3/12	579-7354
8	The Looney Bin	8-bit/ST	3/12	390-9470
7 !	SMART 520 ST	8-bit/ST	3/12	480-9686
7 !	SMART 520 ST	8-bit/ST	3/12	726-4419
5 !	Computer Plus	8-bit/ST	3/12	691-7862

1=TCre, 2=Forem, 4=Michtron ST, 5=Forem ST  
6=Express, 7=FIDO, 8= Proprietary

(\*=Limited hours, != Magic Sac Support)

## S.D.A.C.E. CLASSIFIEDS

Back issues of ANALOG 50 cents apiece  
GENIE MANUAL \$8.00

Call Peter Payne 560-4272

## SSL ENTERPRIZES

Atari Parts and Service

STEVE LAWRENCE 1334 Mimosa Ct.  
Escondido, CA. 92027



## 8-BIT PROGRAM REVIEWS

by Paul Gittins

(Reprinted from Portland Atari Club Computer News)

### VIRTUOSO by VIRTUSONICS

The Virtuoso program is something entirely new in my Atari experience. The program comes with a book nearly a half inch thick. The book is definitely required reading if you want to use the program. Just trying to think of a good way to describe this program is a challenge. Included with the program is a telecom program and a Compuserve startup package. The reason for the communications program is that all the creations made can be uploaded to CIS for use of other Virtuoso users. In fact CIS has set aside a special section just for the Virtuoso files. By now you are probably asking yourself just what I am talking about anyway and what does Virtuoso do. Well, here we go. Virtuoso allows you to create music, graphics, text, and animation. All these can be put together in a single file which when played will give animated graphics and text in time with the music. You are allowed to either create your own or modify the demos that are included. If you have an artistic nature, this is the program for you. It seems to have almost unlimited variations. You don't have to read music to write and play music. I should add that the program allows the use of Joystick, Koala Pad, or Atari touchtablet. These offer mouse-like operation which is clean and positive. I was very impressed with the way the joystick functioned. I was able to write some music just by lining up the cross hairs on the spot where I wanted the note and pressing the button. The package I used was the preliminary version and has since been updated to allow even greater creative freedom.

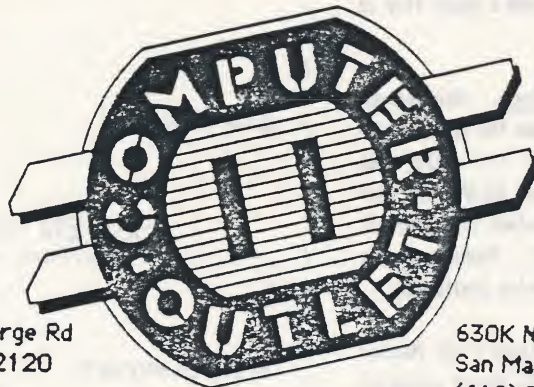
Someone once said that the 8-bit Atari user would never spend more than 30 minutes learning how to use any program. Virtuoso defies this. If you are an artist or musically inclined, or would like to be, this is the program for you. If you would like to sample this program and have access to Compuserve, you can download the player and some of the demos and see for yourself. You may also find some information in the November Antic.

### SOLAR STAR by MICRODAFT

Remember the first time you played PAC MAN and found your quarter gone and game over before you even realized what had happened? That's what happened to me with Solar Star. Fortunately I didn't have to pay a quarter; but I have to admit the action is fast and furious. The idea of the game is to collect enough fuel crystals to allow you to warp to the next level. All you have to do is shoot the disrupters and catch the crystals they carry. When you have 10 crystals, you may move on to the next level. Sound easy? I might add that you are not supposed to run into any of the walls or the disruptors. If you do, you lose 1 unit of fuel for each collision. I really enjoyed this game. The action can be slowed down a bit but remains a challenge. Only twice did I ever get to the 2d level and here I collected about 5 crystals. You play the game on a grid something like Ballblazer, but only requires 1 player. This is 1 game I would want in my personal library!

# CAUTION

## COMPUTER OUTLET MAY BE HABIT FORMING



5945 Mission Gorge Rd  
San Diego, CA 92120  
(619) 282-6200

630K Nordahl Rd  
San Marcos, CA 92069  
(619) 740-0111

6% off all non-sale software and hardware on your next purchase.



Atari 8-Bit Computer  
Programming Languages  
(Ron Miller)

Atari computer owners who are new to the computer world are usually content to work with the built in BASIC programming language. For many people, BASIC is a satisfactory programming language and the only language they ever need. However, many people also hit BASIC's limits or they just get tired of the language. The question then is raised on where to go from BASIC. I can't tell you the path you should follow but I can outline some choices and give some guidelines.

BASIC has two properties that have made it successful. First, it is interactive, that is, you can immediately use it, change it, and use it again. Second, it is usually very well integrated into the computer system. That is, BASIC usually has all the commands for graphics, sound, printing, joy sticks, etc. Thus BASIC hides the underlying computer system details. I use BASIC frequently for small programs. BASIC gets ugly when a) you want functions that BASIC does not provide (like bit manipulations), b) you need speed (like in player missile graphics), c) you want to control something directly in the computer hardware (like the system timers). There are other limits of course, but these are the ones I most frequently run into. Well, what to do about it?

ASSEMBLY LANGUAGE is a possible next step. It is very easy to learn and very hard to use. The 6502 microprocessor used in Atari (and Apple II, Commodore 64) is a very simple microprocessor. Writing 6502 assembly language is easy, interfacing to the Atari operating system is not easy for the uninitiated. There are no nice assembly language commands like GR.8, PRINT, FOR I=1 to 10, A = B + C, etc. You get to do it all. There are tools like macro assemblers which help minimize the pain, but that is an additional thing to learn. However, if you dive fully into assembly language, you must learn these things. There is an alternative. Calling assembly language routines from Atari BASIC is unusually easy. You might try that approach. Use BASIC for all those nasty things like, PRINT, PLOT, math, etc, but write only short assembler routines to do what you can't do in BASIC. If you get interested in this, buy an Atari ASSEMBLER EDITOR cartridge and look at the examples in the appendix of the user manual that comes with the cartridge. The cartridge is a good buy just for the EDITOR and making small assembly routines is easy.

Another possibility is the ACTION language from Optimized System Software. ACTION has many properties similar to Pascal and C. ACTION has true subprograms, pointer variables, the ability to tie into the Atari hardware, etc. ACTION comes with a neat split screen editor and the programming environment is kind of like TURBO Pascal on the IBM PC's (and on CP/M computers). You can edit, compile and run programs right from the editor. And ACTION is fast. This language is a very nice way to learn a bit more about modern programming languages. There are club members who have done extensive work in ACTION, so, if that interests you, you should talk to a fellow club member or arrange to meet at a club meeting and see ACTION in action.

A very cheap alternative is FORTH. FORTH is a very unusual language and (in my opinion) has more general programming power than all the other Atari languages put together. But that power is only available to someone who is willing to invest the time to learn how to use it. FORTH is interactive (but not as interactive as BASIC). FORTH is compiled, but not completely. I have seen many people get excited when they first use FORTH because they can do neat things very simply. But when they try to write more complicated programs they lose their excitement. This is usually because they don't understand enough computer science type things like link lists, arrays, vectored executions, handling raw keyboard inputs, computer memory management, all the things that BASIC does for you behind the scenes. But FORTH is well documented and it is free. Our club library has a copy of the FORTH system.

OTHER LANGUAGES are available. They are usually special purpose, incomplete or are unreasonable to use. PILOT worked very well but it is special purpose (building interactive dialogues). The compiled languages like Pascal and C required you to load an editor, create a program, save the program, load the compiler, compile the program, load a link program, link your program, then load your program and run it. You get the idea. You spend much time watching your disk drive work. That's kind of a slow way to work. I evaluated one version of LISP which would have been a satisfactory tool for learning LISP, but it wasn't well integrated into the Atari.

One more alternative is to upgrade your computer hardware. The above assumes an Atari computer with 64K of memory. We have seen some nice upgrade kits that take the 8-bit memory up to 256K bytes. If you spend money on hardware upgrades you can then use some of these other languages from a RAM disk (you make your computer memory look like a disk drive). Or you can add a hard disk. These things add enough support to your programming environment to make these other languages useful.

I am not aware of everything that has ever been invented for our 8 bit computers so if there is some neat language out there that I have not mentioned, you can poke me about it. However, your thoughts would be much more valuable if you published them in the club newsletter. There are many Atari 8-bit computers out there and the sales of the new 65XE game systems will just increase the count. Eventually some of these owners will be asking the question. Where can I go from BASIC? If you have a good answer, there are others who would like to hear it.

BLINKING CURSOR IN FORTH  
(RON MILLER)

The Atari operating system contains some countdown timers which are decremented 60 times a second during video vertical blank. When any timer hits zero, an interrupt is generated and an interrupt service routine is executed by the operating system. The following program uses one of these timers to periodically change the Atari cursor control register from normal video to inverse video and back.



The following program is written in FORTH. Most FORTH systems have a built in 6502 assembler and the words BLINK and START.BLINK are written in FORTH 6502 assembler. FORTH will allow you to change from FORTH to assembler and back in the same program. (Isn't that neat?) The following FORTH code defines four FORTH words. BLINK is the interrupt service routine which blinks the cursor and resets the countdown timer. START.BLINK loads the timer the first time which starts the blinking. INIT.BLINK stores the address of BLINK into the interrupt service routine vector (228 hex). BC is just a nice short command to execute everything.

Once BC is executed, the Atari cursor will blink without any interference to the main program, which in this case is FORTH. Note that BLINK does not contain the normal "NEXT JMP," to return to FORTH. This is because BLINK is never executed by FORTH. It is executed by the Atari operating system during vertical blank.

You may not really need a blinking cursor, however, that is not the real value of this program. This program demonstrates how easy it is to interface to the Atari hardware using FORTH and use the system timers. This has much value when you program your Atari 400 to water your tomatoes, feed the dog, tune up the car, control home security, answer the phone, and decode morse code from a ham radio.

Anyone wanting a detailed explanation of this code will have to come to an 8-bit meeting.

#### ( BLINKING CURSOR ROUTINES )

( BLINK FLIPS CURSOR VIDEO BIT AND RELOADS THE )  
( COUNTDOWN TIMER WITH 21/60TH OF A SECOND )

```

HEX          ( FOLLOWING NUMBERS ARE IN HEX )
CODE BLINK ( --- )
 02F3 LDA,   ( READ CURSOR CONTROL REGISTER )
 2 # AND,    ( MASK BIT 2 )
 2 # EOR,    ( FLIP BIT 2 )
 02F3 STA,   ( UPDATE CURSOR CONTROL REGISTER )
 15 # LDA,   ( GET READY FOR ABOUT 1/2 SECOND BLINK )
 021A STA,   ( SET TIMER REGISTER )
 RTS,       ( TELL ATARI OS TO RETURN )

```

#### CODE START.BLINK

```

15 # LDA,   ( GET READY FOR ABOUT 1/2 SECOND BLINK )
021A STA,   ( SET OUR TIMER FOR THE FIRST TIME )
NEXT JMP,   ( RETURN TO FORTH )

```

```

: INIT.BLINK ( INSTALLS THE INTERRUPT SERVICE ROUTINE )
  BLINK      ( TAKE THE ADDRESS OF THE BLINK ROUTINE )
 228 ! ;    ( POKE IT INTO THE INTERRUPT VECTOR )

```

```

: BC          ( MAIN PROGRAM )
  INIT.BLINK  ( INSTALL THE ROUTINE )
  START.BLINK ; ( GO AHEAD AND RUN IT )

```

#### 8-BIT CLUB NEWS (RON MILLER)

##### MEETINGS

We held 8-bit club meetings on 7 Jan and 4 Feb. The SDACE board meeting was 3 Jan and the next one is 14 Feb. So you can see we did not have a board meeting between our Jan and Feb club meetings. The board meetings have been changed to the second Sunday of the month and 1 Feb fell on a Monday so there you have it. To add to the confusion, we changed the 8-bit club meeting day to the 2nd Thursday of the month beginning in March 88. So take note. To avoid confusion, here are the meeting dates for the next 6 Atari 8-bit club meetings.

10 March, 14 April, 12 May, 9 June, 14 July, 11 August.

These meeting dates will be better because they don't conflict with the ST Workshop and they miss the holidays like 4 July.

At the Feb meeting it was suggested that we look for a more central meeting place. The Mira Mesa facilities are very nice, but they are hard to find in the dark and kind of far north for many people. Our intention is to find a general meeting place back in San Diego and we can still continue to have workshops and swap meets at the Mira Mesa facility if we like.

##### OCEAN BEACH MEETING

If you have an interest in joining an Atari 8-bit workshop in the Ocean Beach area, call Gil Soares at 222-7436. He is interested in hosting such a meeting and he is very knowledgeable and capably computer user.

##### CLUB BBS

Our BBS has been down due to a blown ATR8000. This ATR8000 has 4 Megabytes of floppy disk attached and the system has not been very reliable. So we are seeking an alternative. We own an 800XL with 256K of memory and what we really need is a hard disk. 10MB hard disks are cheap so we voted to sell the ATR8000 stuff and get a hard disk. This will make a much faster, more capable and reliable system and get our board back on line. Hopefully this will happen within the next couple weeks.

##### CLUB ADS

At the 3 Jan board, we decided to make up some one page ads about our 8-bit and ST computer groups. These ads will be suitable for distribution to vendors who sell Atari computers, including the game systems (65XE, 7800, etc). We believe our newsletter, BBS's and libraries are well worth the annual club dues. But we need to let people know we exist.



# I WROTE SHAREWARE (AND LIVED TO REGRET IT)

by Paul Gardner

(Editor's note: I edited this article from Paul's response to my shareware contribution for GRAFTOOL. I'm as guilty as anyone for not paying shareware authors their due. I can, however, provide a podium for them to vent their gripes. I welcome reader responses from S.D.A.C.E. members and anyone else who reads this.)

From December 1986 to August 1987 I made a bold attempt to support myself with the sole occupation of "Shareware Programmer." For those who snort with derision at the thought, let me make it perfectly clear; during this time I worked as long and as hard (if not longer and harder) than I have ever worked while programming 8 to 5 in a "real" job. I failed in the end; so if what follows betrays some hint of bitterness, you might understand why.

During those months the most promising piece of work I released as shareware was a mathematical graphing program called Graftool. You type in equations and it plots them, no big deal. The shareware agreement I proposed was "If you use the program, send me whatever you think it's worth to you."

I haven't done any work on Graftool since I first released it as shareware. There's plenty of improvement to be made, for sure. Polar coordinates, 3-D graphing, parameterized equations, log-scaled axes, user control over various line styles/colors--all come to mind immediately. It's not that I don't want to make these improvements. I probably will someday.

One thing's for sure, though: The improved Graftool won't be shareware. The response to the original just doesn't justify it. I got one letter. I would have enjoyed getting some feedback from users, even if there wasn't any money in their envelopes.

I don't blame the users. When I'm completely honest with myself, I can even admit that I wouldn't pay money for Graftool. It's a fairly minor program. However, part of the reason I feel that way is probably because I know if I needed such a program, I could whip it up myself. I had expected people less well-versed in programming to be a little more appreciative.

I thought that was the whole beauty of shareware: Bringing users and programmers together and leaving the slick packaging, hot air merchants from Wall Street out in the cold. Golly, half the price of a piece of software you buy in a store goes to the store!

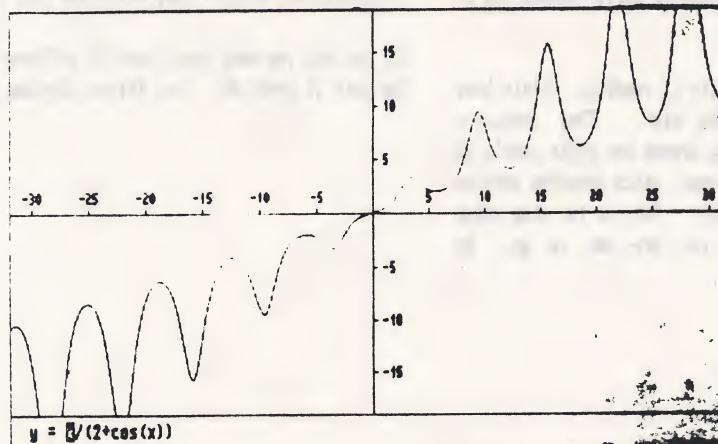
Shareware is a failure. From the programmer's point of view, the only people making money on shareware are the slimeballs who gather it up and sell it as "shareware libraries"--regardless of the fact that many shareware programs specifically condemn this. The few programmers who write amazing shareware hits, like PC-Write for the IBM-PC, would have been better off publishing, anyway.

The users lose, too. Idealistic programmers soon become cynical by the whole mess, they stop producing quality work for shareware and either get a real job (like I did) or start selling their work to publishers. Probably the only things that keeps shareware going is the ever renewed ranks of idealistic newcomers.

I have great admiration for the fortitude of the programmers who don't become cynical, but I really have to question their grip on reality. You only have to read one or two articles from your favorite computer magazine about "tapping the shareware goldmine" to realize that every non-programmer thinks shareware means "free lunch." The people who write these articles--and I'm afraid many who read them--must believe that, for some strange but unimportant reason, programmers just love to provide software for free.

Yeah, sure, I can see these programmers going: "I spent months of my free time pulling out my hair just so I could make this program available to everyone for free." You've got to wonder what motivates a programmer to do such a thing. What makes them tick? The most confounding thing of all is...people like this really exist! That's what "Public Domain" is all about.

Anyway, I've gone on far too long ranting and raving about how shareware stinks. Maybe others have a different story to tell.





## BIG BLUE IN ATARI LAND

by David Delgadillo

PC-DITTO  
AVANT-GARDE SYSTEMS  
381 PABLO POINT DRIVE  
JACKSONVILLE, FL 32225  
PRICE: \$69.00

IBM software is alive and well in the land of Atari. Thanks to PC-Ditto by Avant-Garde Systems, you can emulate an IBM-XT on your ST. This software emulator will allow you to run virtually all IBM software products, either in color or monochrome. Additionally, you may boot DOS from your hard disk if you wish. With a 1 megabyte machine, you have 703K of memory. Double-sided disk drives will give you 720K high capacity disk formats. This article will explore the capabilities of PC-Ditto and give some tips on setting it up.

PC-Ditto 3.1 is the current version. Versions 2.0 and earlier did not allow the use of the monochrome monitor. The current version is also Mega ST compatible, will read the time and date from your clock cartridge upon bootup, and supports both a Generic Mouse (tm) and the Microsoft Mouse (tm). Although a bit slow on graphics redraws, the program shines in the area of applications and utilities.

Disk compatibility is a problem; but with the advent of IBM's PS/2, more programs will be available on 3.5" disks. All of Microsoft's Windows programs are on both 5.25" and 3.5" disks. Some companies will send you a 3.5" disk after proof of purchase. You can use a 5.25" inch external drive to your ST in order to transfer data to a 3.5" disk. In all, compatibility will become less of a problem with the passage of time.

A brief word about PC-DOS. I suggest PC-DOS v.3.2 and v.3.3; especially if you will be using a hard drive. To date, DOS 3.2 and 3.3 are the most compatible with PC-Ditto for accessing all hard disk partitions.

The PC-Ditto manual is massive and rather heavy-handed in its explanations. I would recommend a companion book on DOS to help the novice PC-DOS user. I bought USING PC-DOS by Chris Devoney and it is an invaluable help.

The first step in setting up PC-Ditto is reading. Obtain hard copies of all text files on the PC-Ditto disk. They contain a wealth of information and one can stumble around for quite awhile if they bypass them. Concentrate on the areas which describe setting up the configuration file (CONFIG.SYS). This is the area which causes the most grief. Once past it, you are set to go. My CONFIG.SYS file looks like this:

```
DEVICE=PC_DHD.SYS
BUFFERS=16
DEVICE=DRIVER.SYS /D:0/F:2
DEVICE=DRIVER.SYS /D:1/T:80/S:9/H:1
```

Line 1 recognizes the hard drive and makes partitions D, E, and F available to DOS. Partition C is automatically recognized.

Line 2 sets aside a certain amount of memory for the hard drive to use. It speeds up hard drive use.

Lines 3 and 4 configured my A and B drives and allow me to format 720K disks on drive B, my double-sided drive.

You can create the CONFIG.SYS file with a word processor or from DOS.

Next, click on the PCMENU program found on your PC-Ditto disk. This is where you will set up various parameters for PC-Ditto; such as drives, color/monochrome, screen colors and mouse use. Just experiment. You can always return to it later and resave.

Now, click on the PCDITTO program and get ready to enjoy the IBM world. Tons of software are now ready to flow through your Atari. On the PC-Ditto disk you will find a text file which lists runnable programs. Over 6,000 public domain and shareware programs exist for the IBM, many of which run via PC-Ditto on the ST. If you own a modem, obtain a copy of the shareware program Procomm. Word processors such as PC-Write and New York Word are terrific. Much is out there and it's up to you to find it.

(Editor's notes-Thank you, Dave, for this article. I wish to add that Galaxy, another shareware word processor, has received great notices. See page 142 of the February Computer Shopper for more information.

Just don't forget. ALL shareware developers need their due contributions; whether they write for IBM, Macintosh, or Atari.

So, how did you make your favorite software sit up and do what the ads said it would do? Your fellow Atarians need your help!)



# CX22 TRAKBALL MOD FOR THE ST

A "You-Build-It" Project! By D. E. Wenzelburger, S\*P\*A\*C\*E reprinted from the LACE NEWSLETTER

A while back, after fighting for space on my desk for my mouse, I thought that a trakball, under some circumstances, would be nicer to use. Until Atari Trek I was unsure of which model to use. There I found that an STDIO member had a working setup using the Atari model CX22. The following is a quick (and probably not the only) method to modify the Atari trakball for use on an ST computer.

First a list of parts and equipment needed:

(1) CX22 Trakball. (2) Joystick extender cable (purchased from Radio Shack). (3) 2 feet of 24 or 26 gauge hookup wire. (4) Soldering iron and solder. (5) Wire cutter. (6) Ohm-meter (if possible).

And now on with the show. The purpose of the joystick extender cable is that the exiting cable does not contain enough leads, so we have to replace it. The new cable may be a bit longer than you'd like. I'll leave that up to you. The easiest way to tell you which end to cut off is to plug one end into your ST (turned off of course!!) and cut off the free end. It's also the MALE end that gets cut.

Next unscrew the four screws on the bottom of the CX22, and carefully pull the two halves apart. (Warning, these are sometimes pretty stiff). Remove the plug from the circuit board. Also remove the "Q" ball and the two shafts. Unscrew the screws on the circuit board and remove it.

TABLE 1:

TP1 to TP9  
TP2 to TP12  
TP3 to TP11  
TP4 to TP10

TABLE 2:

1. WHITE	6. ORANGE
2. BLUE	7. RED
3. GREEN	8. BLACK
4. BROWN	9. YELLOW
5. GRAY	

TABLE 3:

BLACK	Gnd	J1-1
(Solder to contact with other BLACK wire)		
RED	+5	J1-2
GREEN	XA	J1-3
BROWN	XB	J1-4
WHITE	YB	J1-5
BLUE	YA	J1-6
ORANGE	L. Button (where RED was)	
YELLOW	R. Button (where RED was)	

See Figures 1 & 2. Carefully cut the foil traces as shown in Figures 1 & 2. Using Table 1, connect the test points show with the hook-up wire. Reinstall circuit board and screws. To be on the safe side, it's a good idea to check out the new cord. I wouldn't trust Radio Shack to always use the same color wires in their cables.

Use Table 2 and Figure 3 to verify that your's is the same as mine. Now back to the trakball case. Remove all of the red wire from the trigger buttons.

Next, carefully pull the black leads on P1. They and their contact should pull from the plug. Cut off the black wire going to the old cable.

The next step may be tricky. I was able to pull the rest of the wires out of P1 without damaging it. If all else fails you can always solder the cable directly to J1.

Carefully strip off enough covering on the new joystick cable so that the wires can reach the trigger switches. Next I applied enough plastic tape around the end of the cable so that when the cover is on there is enough resistance that the cable will not pull out.

Per table 3, connect the new joystick cable. I reused the old plug by cutting some of the plastic dividers down to get a soldering iron on-to the connection. Reconnect plug, replace shafts and "Q" ball, and put cover back on.

The switch on the left hand side must be in the JOYSTICK position for this modification to work. If for some reason it fails to work, carefully recheck the trace cuttings and your P1 connector wiring.

FIGURE 1  
TOP OF BOARD



FIGURE 2  
BOTTOM OF BOARD



FIGURE 3



JOYSTICK CABLE  
PLUG



**IF YOU'VE BEEN PAYING LIST**  
For your Software

**You haven't been shopping at Warners!**

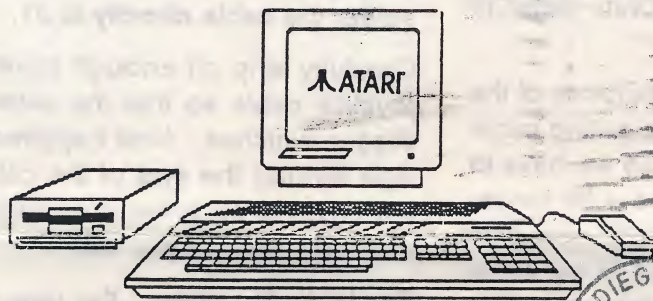
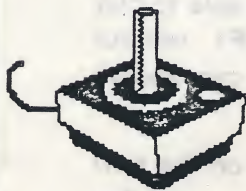
*Professional Midi Studio  
Free Public Domain Library  
Non-Commissioned Sales Staff  
Complete Service Center*

*Full Service  
Free Training  
Mail Order Pricing  
Try Before You Buy*

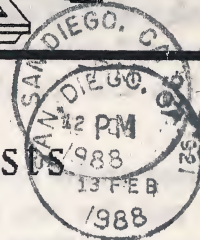
**WARNER**

SOFTWARE  
COMPUTERS  
ENGINEERING

3545 Midway Drive, Suite C  
San Diego, California 92110  
619/224-5090



**San Diego Atari Computer Enthusiasts**  
P.O. Box 203076  
San Diego, CA 92120



**SAN DIEGO, CA 92126**

The February **8-bit** meeting will be Thursday February **4th**, at the Mira Mesa meeting facility, 6:30, and the **ST hands-on/beginners' SIG** will be on the same date and time, at the North Park Rec Center. The **regular ST meeting** will be on February **15th**, at the North Park facility.